**Software Requirements**

**Specification**

**for**

**OrthoVision**

**Version 1.0 approved**

**Prepared by Pereverziev Hlib**

**NURE**

**06.05.2025**

**Історія версій**

|  |  |  |  |
| --- | --- | --- | --- |
| **Дата** | **Опис** | **Автор** | **Коментарі** |
| 19.05.2025 | Версія 1.0 | Переверзєв Гліб | Створення документу |

**Зміст**

Зміст

[1. Вступ 3](#_Toc201681073)

[1.1 Призначення документа 4](#_Toc201681074)

[1.2 Область застосування 4](#_Toc201681075)

[1.3 Цільова аудиторія 5](#_Toc201681076)

[1.4 Терміни, скорочення та абревіатури 6](#_Toc201681077)

[1.5 Огляд документа 7](#_Toc201681078)

[2. Загальний опис 8](#_Toc201681079)

[2.1 Перспектива продукту 8](#_Toc201681080)

[2.2 Функціональність продукту 9](#_Toc201681081)

[2.3 Характеристики користувачів 10](#_Toc201681082)

[2.5 Загальні обмеження 13](#_Toc201681083)

[3. Конкретні вимоги 14](#_Toc201681084)

[3.1 Вимоги до зовнішніх інтерфейсів 14](#_Toc201681085)

[3.1.1 Інтерфейси користувача 14](#_Toc201681086)

[3.1.2 Апаратні інтерфейси 15](#_Toc201681087)

[3.1.3 Програмні інтерфейси 16](#_Toc201681088)

[3.1.4 Комунікаційні інтерфейси 18](#_Toc201681089)

[3.2 Властивості програмного продукту 19](#_Toc201681090)

[3.3 Атрибути програмного продукту 20](#_Toc201681091)

[3.4 Вимоги до бази даних 22](#_Toc201681092)

[3.5 Інші вимоги 24](#_Toc201681093)

# 1. Вступ

Цей документ містить формалізований опис функціональних та нефункціональних вимог до програмного комплексу OrthoVision, який розробляється з метою автоматизації процесів надання медичних послуг у спеціалізованих клініках ортопедичного та офтальмологічного профілю. Документ призначено для розробників програмного забезпечення, аналітиків, тестувальників, керівників проєкту та інших учасників життєвого циклу програмної системи. Його завдання — забезпечити чітке бачення архітектури, функціональності та обмежень системи на ранніх етапах розробки, що дозволить мінімізувати ризики невідповідності очікувань та фактичного результату.

## 1.1 Призначення документа

Метою цього документа є опис і узгодження всіх ключових вимог до системи OrthoVision, яка має забезпечити цифрову взаємодію між пацієнтом, лікарем та адміністративним персоналом медичної установи. Документ орієнтований передусім на технічних спеціалістів, які беруть участь у створенні, тестуванні та підтримці системи. Водночас, він буде корисним і для менеджерів, що ухвалюють рішення, а також для кінцевих користувачів, які бажають зрозуміти можливості програмного забезпечення на етапі планування.

## 1.2 Область застосування

Програмний продукт OrthoVision представляє собою комплексну багатокомпонентну систему, яка охоплює вебзастосунок, мобільний клієнт та серверну частину з REST API, а також підтримку IoT-пристроїв — смарт-окулярів. Система дозволяє здійснювати онлайн-запис до лікаря, вести електронну карту пацієнта, отримувати статистику з пристроїв контролю постави та освітлення, а також адмініструвати діяльність клініки.

З технічної точки зору, OrthoVision складається з:

* мобільного застосунку для пацієнтів (Android), де реалізовано реєстрацію, запис на прийом, перегляд візитів, перегляд персональної медичної карти, а також моніторинг даних з IoT-окулярів;
* вебінтерфейсу лікаря, який дозволяє переглядати календар прийомів, редагувати записи, вказувати діагнози та контролювати стан пацієнтів;
* адміністративного вебінтерфейсу, через який керівництво клініки може додавати лікарів, клініки, користувачів, змінювати їх статус, переглядати загальну статистику;
* серверної частини, яка обробляє запити, керує сесіями, зберігає інформацію в базі даних та відповідає за авторизацію користувачів;
* модуля збору даних, який передає параметри з ESP32-пристроїв через HTTP-запити.

Система не виконує функції електронного рецепта, фінансового обліку або складської логістики. Водночас, архітектура системи дозволяє її розширення в майбутньому за рахунок впровадження додаткових модулів.

Ключові цілі продукту включають:

* зменшення черг та навантаження на медперсонал завдяки онлайн-запису;
* інтеграцію IoT-даних у медичну карту пацієнта;
* покращення взаємодії між клінікою та пацієнтом;
* підвищення зручності та швидкості прийому за рахунок цифровізації основних процедур.

Таким чином, OrthoVision є актуальним рішенням у сфері цифрової медицини, орієнтованим на спеціалізовані заклади, що прагнуть підвищити якість обслуговування пацієнтів і ефективність внутрішніх процесів.

## 1.3 Цільова аудиторія

Цей документ призначений для всіх учасників процесу розробки, впровадження та експлуатації програмного продукту OrthoVision. Основна цільова аудиторія включає:

* Інженери-програмісти, які відповідають за реалізацію серверної частини, вебінтерфейсу та мобільного застосунку, і потребують чітких вимог до функціональності та інтеграції.
* Фахівці з UX/UI-дизайну, які беруть участь у створенні зручних, інтуїтивних інтерфейсів для лікарів, пацієнтів та адміністративного персоналу.
* Тестувальники, відповідальні за перевірку відповідності реалізованого функціоналу вимогам, описаним у цьому документі.
* Менеджери проєкту, які координують розробку, планують етапи, ресурси і контролюють прогрес.
* Клієнти і замовники (наприклад, керівництво клініки або медичної мережі), які зацікавлені у кінцевому функціоналі системи та її відповідності бізнес-вимогам.
* Медичні працівники, зокрема офтальмологи та ортопеди, які будуть використовувати систему у повсякденній практиці для управління записами пацієнтів, ведення карт і перегляду даних з розумних окулярів.
* Адміністративний персонал клінік, який опікується розкладом, управлінням клініками, пацієнтами та користувачами системи.

Документ написаний з урахуванням того, щоб кожен із перелічених учасників міг швидко знайти потрібну інформацію, зрозуміти очікувану поведінку системи і працювати відповідно до єдиної специфікації.

## 1.4 Терміни, скорочення та абревіатури

У цьому підрозділі подано основні терміни, абревіатури та скорочення, які використовуються в даному документі. Їхнє тлумачення є важливим для коректного розуміння вимог до програмного забезпечення OrthoVision.

OrthoVision – назва розроблюваної програмної системи для офтальмологічних та ортопедичних клінік, що забезпечує електронний запис, ведення медичних карт та моніторинг фізіологічних даних пацієнта з IoT-пристроїв.

IoT (Internet of Things) – технологія, яка передбачає підключення фізичних пристроїв до інтернету з метою обміну даними.

ESP32 – мікроконтролер з підтримкою Wi-Fi та Bluetooth, що використовується в системі для передачі сенсорних даних з розумних окулярів.

JWT (JSON Web Token) – стандарт токенів, який використовується для безпечної аутентифікації та авторизації користувачів у вебсередовищі.

API (Application Programming Interface) – інтерфейс прикладного програмування, що дозволяє клієнтам (веб або мобільним) взаємодіяти з серверною частиною системи.

REST (Representational State Transfer) – архітектурний стиль, який використовується при реалізації API системи.

UX/UI – користувацький досвід (User Experience) і користувацький інтерфейс (User Interface), що визначають зручність і логіку взаємодії з системою.

Admin Panel – вебінтерфейс для адміністративного керування користувачами, клініками, лікарями та пацієнтами.

Patient Card – електронна карта пацієнта, що зберігає особисту інформацію, історію прийомів, діагнози та показники з IoT-пристрою.

Appointment – запис на прийом до лікаря із зазначенням дати, часу та статусу.

Doctor Dashboard – інтерфейс лікаря, що дозволяє переглядати календар візитів, обробляти записи та додавати діагнози.

Цей перелік не є вичерпним і може бути доповнений впродовж подальшої розробки документа. Для зручності всі терміни вживаються у своїй усталеній формі по тексту без додаткових розшифровок.

## 1.5 Огляд документа

Цей документ специфікації програмних вимог (SRS) містить повний опис функціональності, архітектури, обмежень і критеріїв якості для програмного забезпечення OrthoVision — системи для управління процесами обслуговування пацієнтів в ортопедичних та офтальмологічних клініках із підтримкою IoT-пристроїв.

Документ структуровано таким чином, щоб охопити всі етапи розробки програмної системи — від загального опису призначення до конкретних функціональних та нефункціональних вимог. У ньому міститься аналіз предметної області, опис інтерфейсів, архітектури, очікуваної поведінки, технічних обмежень і середовища, в якому працюватиме продукт.

Дані, викладені у документі, слугують спільною точкою для розуміння системи між розробниками, дизайнерами, тестувальниками, замовниками та кінцевими користувачами.

У наступних розділах подано:

* загальні відомості про продукт, включаючи його цілі, призначення, функціонал і особливості застосування;
* докладні функціональні вимоги для кожної частини системи (веб та мобільна платформи);
* технічні деталі щодо взаємодії з IoT-пристроєм (розумними окулярами);
* нефункціональні вимоги, пов’язані з продуктивністю, безпекою, доступністю та масштабованістю;
* діаграми та моделі, що ілюструють архітектуру системи.

# 2. Загальний опис

## 2.1 Перспектива продукту

Програмна система OrthoVision є самостійним програмним продуктом, що реалізує клієнт-серверну архітектуру та включає веб-інтерфейс для лікарів і адміністраторів, мобільний застосунок для пацієнтів, а також підтримку IoT-пристроїв — смарт-окулярів для відстеження нахилу голови та рівня освітлення.

Система взаємодіє з базою даних, забезпечуючи зберігання, обробку та доступ до інформації про користувачів, записи на прийом, медичні діагнози та статистику з сенсорів. Комунікація між клієнтськими інтерфейсами та сервером відбувається через REST API, реалізоване за допомогою фреймворку Fiber (Go).

Вебінтерфейс поділено на дві ключові частини: функціонал адміністратора (керування клініками, лікарями, пацієнтами, статистикою) та інтерфейс лікаря (робота з прийомами, встановлення діагнозів, перегляд історії хвороб). Мобільний застосунок, у свою чергу, орієнтований на пацієнта, надаючи можливість запису до лікаря, перегляду картки пацієнта, спостереження за особистими даними зі смарт-окулярів і перегляду візитів.

IoT-пристрій (смарт-окуляри) наразі моделюється через емулятор ESP32 і передає показники освітлення та положення голови на сервер, що дозволяє лікарю отримувати додаткові дані щодо умов, у яких перебуває пацієнт.

Продукт розроблено з урахуванням подальшого масштабування — як у розрізі кількості користувачів, так і розширення функціоналу системи. У майбутньому можливе додавання нових ролей, спеціалізацій клінік, аналітичних панелей, повноцінної інтеграції з фізичними сенсорами та системами обміну медичними даними (EHR/EMR).

## 2.2 Функціональність продукту

Програмна система OrthoVision реалізує низку функціональних можливостей, орієнтованих на різні типи користувачів: пацієнтів, лікарів та адміністраторів клініки.

Автентифікація та ролі:

* F1: Реєстрація та вхід користувачів через email/пароль із використанням JWT.
* F2: Призначення ролей користувачам (пацієнт, лікар, адміністратор) для контролю доступу.
* F3: Розмежування прав доступу на клієнтському та серверному рівнях.

Пацієнтський мобільний застосунок:

* F4: Перегляд списку лікарів за спеціалізацією.
* F5: Запис на прийом з вибором дати та часу.
* F6: Перегляд майбутніх та завершених візитів.
* F7: Доступ до особистої картки пацієнта.
* F8: Перегляд статистики зі смарт-окулярів.
* F9: Можливість звернення до клініки безпосередньо з мобільного застосунку.

Вебінтерфейс для лікарів:

* F10: Перегляд запланованих візитів у вигляді календаря з можливістю фільтрації та деталізації.
* F11: Відкриття окремих записів на прийом.
* F12: Додавання або редагування діагнозів.
* F13: Зміна статусу прийому.
* F14: Перегляд статистики з IoT-пристрою пацієнта для прийняття обґрунтованих рішень.

Інтерфейс адміністратора:

* F15: Керування структурою клініки: додавання та редагування інформації про лікарів, пацієнтів, клініки.
* F16: Перегляд загальної статистики записів, користувачів та активності.
* F17: Відображення змін у реальному часі завдяки інтеграції з серверною частиною через REST API.

Серверна логіка та зберігання даних:

* F18: Обробка та зберігання медичних записів.
* F19: Обробка запитів на запис до лікаря.
* F20: Прив’язка сенсорних показників (IoT) до конкретного пацієнта.
* F21: Генерація даних для графіків і статистики.
* F22: Архітектура з урахуванням масштабування в майбутньому.

## 2.3 Характеристики користувачів

Користувачами системи OrthoVision є пацієнти, лікарі та адміністратори клінік, які взаємодіють із платформою для забезпечення ефективного медичного обслуговування та управління клінічними даними. Для комфортної роботи з системою користувачам бажано мати базові навички роботи з мобільними або веб-застосунками відповідно до їх ролі.

Категорії користувачів.

Пацієнт:

* використовує мобільний застосунок для запису на прийом і перегляду медичної інформації;
* має базові навички роботи зі смартфоном або планшетом;
* потребує інтуїтивного інтерфейсу та швидкого доступу до особистих даних і статистики зі смарт-окулярів;
* може звертатись до клініки через застосунок безпосередньо.

Лікар:

* працює через вебінтерфейс на комп’ютері або ноутбуці;
* має середній або високий рівень комп’ютерної грамотності;
* використовує систему для перегляду запланованих візитів, ведення медичних записів, внесення діагнозів і зміни статусів прийому;
* аналізує статистику пацієнта, отриману з IoT-пристроїв.

Адміністратор:

* здійснює управління даними про лікарів, пацієнтів та клініки через вебінтерфейс;
* має повний доступ до системи та правомочність змінювати конфігурації;
* відповідає за підтримку актуальності інформації і цілісність бази даних;
* володіє навичками роботи з веб-застосунками та розуміє основи адміністрування систем.

**2.4 Загальні обмеження**

Програмна система OrthoVision розроблена з урахуванням сучасних технологічних стандартів, однак має певні обмеження, які слід враховувати при її впровадженні, експлуатації та подальшому розвитку.

Перш за все, система залежить від наявності стабільного інтернет-з’єднання. Вся взаємодія між клієнтськими застосунками (мобільним та вебінтерфейсом) і серверною частиною здійснюється через REST API з використанням протоколу HTTP(S). Відсутність або низька якість мережевого зв’язку може призвести до затримок у передачі даних, неможливості завантаження інформації або навіть втрати сесії користувача, що безпосередньо впливає на користувацький досвід та ефективність роботи.

Друге обмеження пов’язане з платформною сумісністю мобільного застосунку. Орієнтованість на сучасні версії операційних систем Android та iOS визначає мінімальні системні вимоги до пристроїв користувачів. Застарілі пристрої з обмеженими апаратними ресурсами або старими версіями ОС можуть некоректно відображати інтерфейс або не підтримувати всі функціональні можливості застосунку, що обмежує коло користувачів.

Третім суттєвим обмеженням є інтеграція з IoT-пристроями, такими як смарт-окуляри та сенсорні модулі. Система наразі обробляє та аналізує обмежений набір сенсорних параметрів (наприклад, рівень освітлення, положення голови), що не охоплює всі аспекти моніторингу здоров’я пацієнта. Розширення спектру вимірюваних даних потребує додаткової розробки та оновлення апаратного забезпечення.

З точки зору безпеки, система впроваджує базові механізми захисту, такі як автентифікація користувачів за допомогою JWT та розмежування доступу за ролями. Проте відсутність більш складних технологій шифрування даних у базі, багатофакторної аутентифікації або інтеграції з системами управління ідентифікацією може бути недостатньою для роботи у медичних установах з підвищеними вимогами до інформаційної безпеки.

Архітектура застосунку має монолітний характер, що спрощує початкове розгортання і підтримку, але водночас накладає обмеження на масштабованість та гнучкість розвитку системи. Відсутність мікросервісної структури може ускладнювати паралельну розробку та впровадження нових модулів без впливу на стабільність роботи інших компонентів.

Щодо локалізації, поточна реалізація підтримує лише дві мови — українську та англійську. Це обмежує можливість використання системи у міжнародних або багатомовних установах, де потрібна підтримка додаткових мов для комфортної роботи користувачів.

Нарешті, система не передбачає роботу в офлайн-режимі, що може бути критичним у випадках тимчасової втрати інтернет-з’єднання або при використанні у віддалених регіонах з нестабільною мережею.

Врахування вказаних обмежень є важливим для планування подальших етапів розвитку OrthoVision, а також для вибору сфери застосування, де система буде найбільш ефективною.

## 2.5 Загальні обмеження

У процесі розробки програмної системи OrthoVision було враховано низку припущень, які визначають очікувані умови функціонування системи, а також залежності, що впливають на її стабільну роботу та розвиток.

Передбачається, що кінцеві користувачі — пацієнти, лікарі та адміністратори — мають мінімальні необхідні технічні навички для взаємодії із системою. Пацієнти повинні вміти користуватися мобільними застосунками на Android-пристроях, лікарі та адміністратори — працювати у вебінтерфейсі через сучасний браузер (Google Chrome, Firefox, Edge тощо). Очікується, що доступ до інтернету є сталим і достатнім для коректної роботи застосунків.

Розгортання серверної частини припускає наявність хостингу або локального серверного середовища з підтримкою бази даних PostgreSQL, середовища виконання для Go (Golang) та можливості обробки HTTP-запитів. Поточна версія не використовує Docker або Kubernetes, однак допускається інтеграція таких технологій у майбутньому.

Система передбачає, що всі користувачі реєструються з унікальними електронними адресами, а збереження ідентифікаційних токенів відбувається у захищеному сховищі (у мобільному додатку — через EncryptedSharedPreferences, у вебверсії — у localStorage або cookies). Безпека передачі даних забезпечується використанням HTTPS-протоколу.

Функціональність, пов’язана з обробкою сенсорних даних, базується на коректній роботі IoT-пристроїв (зокрема, смарт-окулярів з мікроконтролером ESP32) та їх здатності передавати дані у форматі, узгодженому з сервером. Передбачається, що пристрої вже налаштовані, підключені до Wi-Fi і мають доступ до мережі.

Залежність від сторонніх бібліотек, таких як React, Next.js, Jetpack Compose, GORM, Fiber, та інших інструментів, створює потенційні ризики сумісності при оновленні версій. Також у системі частково застосовуються сторонні рішення для стилізації (CSS-модулі), багатомовності (i18next), роботи з JWT (golang-jwt/jwt) і HTTP-клієнтів (axios, Retrofit). Усі вони вимагають підтримки та тестування при переході до нових релізів.

Нарешті, передбачається, що база даних підтримується актуальною та цілісною, а резервне копіювання даних здійснюється адміністраторами клініки відповідно до внутрішніх політик. Відсутність таких заходів може призвести до втрати критично важливої медичної інформації.

# 3. Конкретні вимоги

## 3.1 Вимоги до зовнішніх інтерфейсів

### 3.1.1 Інтерфейси користувача

Програмна система OrthoVision реалізує кілька користувацьких інтерфейсів, кожен з яких адаптований до функціональних потреб конкретної ролі користувача. Основну взаємодію з системою здійснюють три категорії користувачів: пацієнти, лікарі та адміністратори клінік. Кожен із цих типів має окремий інтерфейс, розроблений відповідно до принципів зручності використання (usability), доступності (accessibility) та відповідності функціональним сценаріям.

Інтерфейс пацієнта реалізовано у вигляді мобільного застосунку, розробленого на базі технологій Jetpack Compose для Android. Користувач після авторизації отримує доступ до особистого кабінету, де може здійснити запис на прийом, переглянути історію візитів, відкрити карту пацієнта з попередніми діагнозами, а також переглянути статистику зі смарт-окулярів. Інтерфейс побудований на основі концепції bottom navigation, де кожен розділ представлений окремою вкладкою. Додатково реалізовано можливість звернення до клініки за допомогою спеціального екрана зв’язку.

Інтерфейс лікаря реалізовано як вебзастосунок на основі Next.js з використанням адаптивного дизайну для забезпечення комфортної роботи на десктопах. Лікар отримує доступ до особистого календаря із запланованими візитами, має змогу відкривати кожен прийом у вигляді окремої картки, додавати або редагувати діагнози, закривати візити, а також переглядати графіки зі смарт-окулярів пацієнта. Основні елементи інтерфейсу — панель навігації, сторінка прийомів (appointments), вікно редагування картки пацієнта та сторінка статистики. Всі зміни відображаються в реальному часі завдяки інтеграції з API.

Інтерфейс адміністратора також побудований як частина вебінтерфейсу, з розділеним доступом на окремі сторінки для керування пацієнтами, лікарями, клініками, а також перегляду загальної статистики. Для кожного ресурсу реалізовано повний CRUD-функціонал. Інтерфейс підтримує можливість фільтрації, сортування та пошуку інформації в таблицях. Передбачено обмеження доступу до адміністративних сторінок згідно з роллю користувача, яка визначається після авторизації.

Інтерфейси усіх типів користувачів підтримують локалізацію українською та англійською мовами. Для забезпечення цілісності користувацького досвіду застосовуються єдині стилістичні підходи: колірна схема, шрифти, компоненти керування. Враховано принципи доступності — контрасти, розміри кнопок, логічна послідовність фокусів, що особливо важливо у сфері медичних сервісів.

### 3.1.2 Апаратні інтерфейси

Програмна система OrthoVision забезпечує підтримку роботи у всіх сучасних браузерах, що встановлені на персональних комп’ютерах та мобільних пристроях. Зокрема, передбачається підтримка десктопних операційних систем Windows, macOS та Linux. Для мобільних пристроїв підтримується платформа Android версії 9 і вище. Мінімальна рекомендована ширина екрану складає 360 пікселів, що забезпечує коректне відображення інтерфейсу на різних пристроях.

Серверна частина:

Серверна частина системи розгортається на фізичних або віртуальних серверах із встановленим середовищем виконання Go (Go runtime environment). Для зберігання та обробки даних використовується реляційна система управління базами даних PostgreSQL. Серверна частина реалізує REST API, що забезпечує обробку запитів від клієнтських застосунків і IoT-пристроїв. Система розрахована на середнє навантаження, що передбачає одночасну роботу кількох десятків сенсорів. Також передбачена безпосередня взаємодія з мікроконтролерами ESP32 через HTTP-запити.

Інтерфейси IoT-пристроїв

До складу апаратної частини входять мікроконтролери ESP32, які підключаються до Wi-Fi мережі для передачі даних. Для збору інформації використовуються сенсори MPU6050, що виконують функції акселерометра та гіроскопа, а також аналоговий датчик освітлення (LDR). Дані передаються на сервер у вигляді періодичних HTTP POST-запитів у форматі JSON, що забезпечує своєчасний та надійний обмін інформацією.

### 3.1.3 Програмні інтерфейси

У межах програмної системи OrthoVision реалізовано комплекс зовнішніх програмних інтерфейсів, що забезпечують взаємодію між користувачами, серверною логікою та IoT-пристроями. Архітектура побудована з дотриманням принципів розділення відповідальностей, безпечного доступу до ресурсів і масштабованості в майбутньому.

Механізм автентифікації базується на технології JSON Web Token (JWT), що дозволяє реалізувати безпечну авторизацію без потреби зберігати сесії на сервері. Після успішного входу користувач отримує токен доступу, який надалі передається в кожному запиті через HTTP-заголовок Authorization: Bearer <token>. Система перевіряє валідність токена, декодує його вміст і визначає права доступу згідно з роллю користувача (адміністратор, лікар або пацієнт). Передача всіх даних між клієнтами та сервером здійснюється з використанням захищеного протоколу HTTPS.

Серверна частина реалізована на мові програмування Go з використанням вебфреймворку Fiber, що забезпечує обробку запитів відповідно до REST-підходу. Підтримуються всі основні HTTP-методи: GET, POST, PUT та DELETE. Обмін даними між клієнтом і сервером здійснюється у форматі JSON. Ключові ендпоінти охоплюють авторизацію користувачів (/auth/login, /auth/register), керування даними про користувачів, лікарів і пацієнтів (/users, /doctors, /patients), взаємодію з записами на прийом (/appointments, /appointment-times), а також обробку даних, отриманих від смарт-окулярів (/smart-glasses).

Особливу роль у системі відіграє інтеграція з IoT-пристроями — зокрема, із мікроконтролерами ESP32, до яких підключені сенсори освітлення та положення голови. Пристрої періодично надсилають HTTP POST-запити до відповідного ендпоінту сервера (/smart-glasses). Кожне повідомлення містить у форматі JSON інформацію про рівень освітлення (в lux), кут нахилу голови (в градусах) та ідентифікатор користувача. Отримані дані використовуються для формування статистики, яка допомагає лікарю оцінити умови навколишнього середовища та ергономіку пацієнта між візитами.

Для зберігання даних використовується реляційна база даних PostgreSQL. Взаємодія із нею реалізована за допомогою ORM-бібліотеки GORM, що забезпечує зручну абстракцію над SQL-запитами. Структура бази даних побудована відповідно до предметної області й охоплює такі таблиці:

* Таблиця users зберігає облікові записи всіх користувачів системи. У ній зберігається інформація про email, хешований пароль, а також роль користувача, що визначає рівень доступу до функціоналу.
* Таблиця doctors містить розширені дані про лікарів — спеціалізацію, ПІБ, контактну інформацію. Вона зв’язується з таблицею users через зовнішній ключ.
* Таблиця patients акумулює персональну інформацію про пацієнтів: ім’я, вік, стать та інші дані, які можуть бути використані при медичному обслуговуванні. Зв’язок із таблицею users реалізується аналогічно до лікарів.
* Таблиця appointments відображає заплановані, завершені або скасовані візити пацієнтів до лікарів. Кожен запис містить дані про учасників візиту, дату, час, статус та діагноз, встановлений лікарем.
* Таблиця appointment\_times виконує функцію зберігання доступних часових слотів. Вона дозволяє формувати розклад лікаря і забезпечує логіку вибору часу під час створення запису на прийом. Кожен слот прив’язаний до конкретного лікаря та дати.
* Таблиця smart\_glasses\_data містить усі дані, отримані від IoT-пристрою. Зокрема, це значення освітлення (eye\_strain), кут нахилу голови (posture\_angle) та мітка часу надходження (timestamp). Запис містить зв’язок із конкретним користувачем, що дозволяє забезпечити індивідуальну аналітику по кожному пацієнту.

Така структурна організація даних дозволяє ефективно оперувати як адміністративною інформацією (облікові записи, ролі, розклад), так і медичною (візити, діагнози, сенсорні показники). У поєднанні з API та авторизаційною логікою, ця модель забезпечує цілісність, безпеку й аналітичну цінність даних у системі OrthoVision.

### 3.1.4 Комунікаційні інтерфейси

Функціонування програмної системи OrthoVision передбачає постійну взаємодію між окремими її компонентами через мережеві протоколи. Основним середовищем для обміну даними є локальна або глобальна мережа з використанням протоколу HTTP/HTTPS. Обмін інформацією відбувається за допомогою форматованих запитів і відповідей у форматі JSON, що є галузевим стандартом для RESTful API.

З'єднання між клієнтськими додатками (веб-інтерфейс для лікарів і адміністраторів, мобільний застосунок для пацієнтів) та серверною частиною здійснюється за допомогою HTTP(S)-запитів до публічно доступного API. Для безпеки передавання чутливих даних (облікова інформація, медичні записи, дані з сенсорів) використовується захищений протокол HTTPS, який забезпечує шифрування трафіку за допомогою TLS.

Передбачено двосторонню взаємодію між IoT-пристроями (смарт-окулярами на базі ESP32) та сервером. Смарт-окуляри ініціюють вихідні HTTP POST-запити до відповідного маршруту API (/smart-glasses), надсилаючи дані про освітленість та нахил голови у вигляді JSON-об’єкта. Пристрої використовують вбудовані засоби з’єднання Wi-Fi і надсилають дані періодично або у разі виявлення порушення умов.

Усі API-запити, що потребують ідентифікації користувача, містять заголовок авторизації у форматі Authorization: Bearer <JWT>. Серверна частина проводить валідацію токена перед обробкою запиту. У разі виявлення помилки (некоректний токен, закінчення терміну дії, відсутність прав) сервер повертає відповідь із HTTP-кодами помилок (401 Unauthorized, 403 Forbidden тощо).

Для забезпечення коректної взаємодії клієнтів із сервером передбачена кросплатформеність — запити можуть надходити як із браузера, так і з мобільного додатка. Підтримується робота через проксі (наприклад, ngrok) під час розробки або тестування.

Таким чином, комунікаційні інтерфейси OrthoVision реалізовано відповідно до сучасних принципів веброзробки та IoT-взаємодії, із фокусом на безпеку, сумісність і масштабованість.

## 3.2 Властивості програмного продукту

Програмну систему OrthoVision розроблено як багаторівневу клієнт-серверна систему, що забезпечує централізоване збирання, зберігання, обробку та візуалізацію медичних даних, зокрема параметрів постави та освітлення, отриманих зі смарт-окулярів. Властивості програмного забезпечення базуються на потребах точного моніторингу стану пацієнтів, надійної автентифікації користувачів, а також масштабованості і гнучкості у подальшому розвитку системи.

Система складається з незалежних компонентів:

* серверна частина, реалізована на мові Go з використанням фреймворку Fiber, що відповідає за бізнес-логіку, обробку REST API запитів, автентифікацію через JWT, та взаємодію з базою даних;
* база даних PostgreSQL — для структурованого та надійного зберігання інформації про користувачів, прийоми, медичні записи та показники з IoT-пристроїв;
* веб-інтерфейс (Next.js + React) — для адміністрування системи, управління даними лікарів, пацієнтів та адміністраторів, а також перегляду і редагування прийомів і діагнозів;
* мобільний застосунок (Kotlin + Jetpack Compose) — для пацієнтів, що забезпечує зручний доступ до запису на прийом, перегляду власної медичної картки та статистики зі смарт-окулярів;
* IoT-рівень (ESP32 + сенсори MPU6050, LDR) — для збору даних про нахил голови пацієнта та рівень освітлення в реальному часі.

Дана архітектура дозволяє підтримувати високу надійність роботи системи, забезпечує захищений обмін інформацією, а також є масштабованою, що відкриває можливості для подальшого розширення функціональності, інтеграції нових пристроїв та адаптації до змін у вимогах медичної практики.

## 3.3 Атрибути програмного продукту

Програмна система OrthoVision розроблялася з урахуванням сучасних вимог до медичних інформаційних систем, що використовуються в контексті клінічної практики, зокрема офтальмологічної та ортопедичної. Нижче наведено ключові властивості, які визначають функціональні та нефункціональні характеристики системи.

Надійність.

Система гарантує збереження та коректне відображення даних у разі стандартних операцій. Завдяки використанню перевірених інструментів (PostgreSQL, GORM, Fiber) та застосуванню механізмів валідації на рівні клієнта і сервера, мінімізується ризик помилок при введенні та обробці інформації.

Безпека.

OrthoVision використовує авторизацію на основі JWT-токенів, що дозволяє забезпечити чітке розмежування доступу відповідно до ролі користувача (адміністратор, лікар, пацієнт). Дані передаються через HTTPS, що унеможливлює перехоплення або підміну інформації в процесі транспортування. Вразливості до XSS або CSRF усунуті завдяки суворій політиці автентифікації та обмеженням на стороні бекенду.

Зручність у користуванні.

Інтерфейс мобільного додатку оптимізований для кінцевого користувача — пацієнта, із врахуванням принципів UX-дизайну: інтуїтивна навігація, адаптивність до різних розмірів екранів, використання зрозумілих іконок і підказок. Вебінтерфейс для лікарів і адміністраторів організований на основі дашбордів, таблиць і фільтрів, що спрощує пошук, аналіз і модифікацію медичних даних.

Продуктивність.

Усі запити до серверної частини обробляються асинхронно, що забезпечує швидкий відгук інтерфейсу та дозволяє обробляти десятки одночасних з’єднань без істотного зниження продуктивності. База даних оптимізована за допомогою індексів та реляційних зв’язків, що забезпечує ефективну роботу навіть при збільшенні кількості користувачів і записів.

Портативність та кросплатформеність.

Застосунок підтримує мобільні пристрої на базі Android 9.0 і вище, а також усі сучасні браузери на ОС Windows, Linux та macOS. Серверна частина не має жорстких вимог до хостингу — вона може бути розгорнута на будь-якому середовищі, що підтримує Go та PostgreSQL (локальні сервери, VPS, хмарні провайдери).

Масштабованість.

Архітектура системи дозволяє легко інтегрувати нові модулі, типи пристроїв або ролі користувачів. Наприклад, можна додати інтерфейс для медсестри, модуль сповіщень або розширити базу діагнозів. Дані з сенсорів можуть надходити з різних джерел одночасно, що відкриває можливості для паралельного моніторингу великої кількості пацієнтів.

Модульність.

Кодова база структурована відповідно до принципу розділення відповідальностей: логіка контролерів, моделей, роутів, middleware і утиліт розділена за окремими модулями. Це значно спрощує підтримку, тестування та розширення функціоналу.

Можливість обробки IoT-даних.

Однією з унікальних властивостей є інтеграція з пристроєм смарт-окулярів, що дозволяє відслідковувати показники освітлення та положення голови пацієнта. Дані надходять у реальному часі та зберігаються в базі даних для подальшого аналізу лікарем.

У сукупності ці властивості дозволяють програмній системі OrthoVision бути гнучкою, безпечною та придатною до впровадження у сучасних клінічних установах, які прагнуть цифровізації процесів і впровадження IoT-рішень у медичну практику.

## 3.4 Вимоги до бази даних

База даних є ключовим компонентом програмної системи OrthoVision, що забезпечує збереження, структуроване організування та швидкий доступ до великого обсягу медичних даних, а також інформації, отриманої від IoT-пристроїв. Відповідно, вимоги до бази даних визначаються як з точки зору функціональності, так і з огляду на продуктивність, надійність та безпеку.

Тип системи керування базою даних:

Використовується реляційна система управління базами даних PostgreSQL, що підтримує складні запити, транзакції, індексацію, реплікацію та забезпечує високу надійність збереження даних.

Структура та цілісність даних:

База даних повинна містити таблиці, які чітко відображають модель предметної області системи, включаючи сутності користувачів, ролей, записів на прийом, медичних діагнозів, а також дані з пристроїв смарт-окулярів. Обов’язковим є дотримання нормалізації для уникнення надмірного дублювання даних, забезпечення зв’язків між таблицями (через зовнішні ключі) та цілісності даних.

Продуктивність:

База даних повинна забезпечувати швидкий доступ до даних навіть при зростанні обсягів інформації, тому необхідне застосування індексів на полях, які часто використовуються в умовах пошуку, фільтрації або сортування. Запити, що виконуються найчастіше (наприклад, отримання розкладу прийомів лікаря чи історії вимірів пацієнта), мають бути оптимізовані.

Безпека:

Забезпечення конфіденційності та цілісності даних реалізується через налаштування прав доступу до бази даних, використання аутентифікації і шифрування збережених даних за потреби. Адміністративний доступ до бази даних обмежений, а операції з медичними даними ведуться з журналюванням.

Масштабованість та розширюваність:

Структура бази даних повинна бути спроєктована таким чином, щоб у подальшому можна було легко додавати нові сутності або атрибути без суттєвих змін у поточних таблицях. Це забезпечить гнучкість і можливість інтеграції додаткових функціональних модулів.

Резервне копіювання та відновлення:

Повинна бути реалізована регулярна система резервного копіювання даних із можливістю швидкого відновлення у випадку збоїв або втрати інформації.

Інтеграція з серверною частиною:

База даних інтегрується із сервером через ORM-бібліотеку GORM, що спрощує взаємодію, забезпечує типобезпечний доступ до даних і дозволяє знижувати кількість ручних SQL-запитів.

Усі ці вимоги сприяють створенню надійної, безпечної та продуктивної бази даних, що є фундаментом для стабільної роботи системи OrthoVision в умовах медичного середовища.

## 3.5 Інші вимоги

Окрім функціональних і технічних вимог, до програмної системи OrthoVision пред’являються додаткові вимоги, що забезпечують її якісну та безпечну експлуатацію, а також зручність для користувачів.

Вимоги до безпеки:

Система повинна гарантувати захист персональних даних пацієнтів відповідно до чинного законодавства про захист медичної інформації. Всі з’єднання між клієнтами та сервером мають бути захищені за допомогою протоколу HTTPS із використанням TLS. Авторизація та аутентифікація користувачів здійснюється через безпечний механізм JWT-токенів із обмеженим терміном дії. Забороняється зберігання паролів у відкритому вигляді.

Вимоги до продуктивності:

Система повинна забезпечувати швидку реакцію інтерфейсів користувачів, не перевищуючи часу відповіді сервера 200 мс при стандартному навантаженні. Обробка даних з IoT-пристроїв повинна відбуватись у режимі близькому до реального часу з затримкою не більше 1 секунди.

Вимоги до надійності:

Система має забезпечувати високу доступність і стійкість до відмов. У разі виникнення помилок слід передбачити механізми логування і повідомлення адміністраторів для оперативного реагування.

Вимоги до масштабованості:

Архітектура системи передбачає можливість горизонтального масштабування серверної частини та розширення функціоналу без суттєвого впливу на існуючу структуру даних.

Вимоги до зручності користування:

Інтерфейси користувачів повинні бути інтуїтивно зрозумілими, адаптивними для різних пристроїв та доступними для людей із обмеженими можливостями. Передбачена локалізація інтерфейсів українською та англійською мовами.

Вимоги до технічної підтримки:

Система має підтримувати оновлення та обслуговування без тривалих простоїв, а також надавати можливість моніторингу стану серверів та бази даних у режимі реального часу.

Вимоги до документації:

Для користувачів та адміністраторів повинна бути розроблена детальна технічна документація, що містить опис функціональності, інструкції з експлуатації та усунення типових проблем.